

## REMARKS

In connection with the RCE filed herewith, Applicant replies to the Final Office Action dated May 12, 2009, within 2 months. Claims 1-3, 6-28, 30-36 and 39 are pending in the application and the Examiner rejects claims 1-3, 6-28, 30-36 and 39. Support for the amendments may be found in the originally-filed specification, claims, and figures. No new matter has been introduced by the amendments. Applicant respectfully requests reconsideration of this application.

Applicant thanks the Examiner for the interview on June 4, 2009 and the discussion of the combination of the Aghera and Kraml references with respect to whether or not the references are properly combinable and the discussion of potential claim amendments.

The Examiner rejects claims 1-3, 6-24, 26-28, 31-36 and 39 under 35 U.S.C. 103(a) as being unpatentable over Aghera et al., U.S. Publication No. 2004/0098715 A1 ("Aghera"), in view of Kraml et al., U.S. Patent No. 6,141,683 ("Kraml"). Applicant respectfully disagrees with these rejections, but Applicant amends the claims (without prejudice or disclaimer) in order to further clarify the patentable aspects of the claims and to expedite prosecution.

Aghera discloses "an architecture for over the air management of software on a wireless device [that] includes a software architecture supporting software patches, including secure downloading of software from a data network and robust installation of the same on the wireless device" (Abstract). Significantly, Aghera discloses that "[o]nce the patch installation is completed, the stored patch data is deleted..." (para. 55). On the other hand, Kraml maintains separate versions of software in storage: "[f]irst memory 340 also advantageously contains at least a *new version* of the desired software application and at least one *older version* of the software application." (p. 5, ln. 5-8). "Second memory 350 is advantageously a volatile memory...that is of sufficient capacity to store any *one version* of the software application." (p. 5, ln. 27-29; emphasis added).

Therefore, Aghera teaches against Kraml because Aghera only maintains a single executable software program (even if Aghera stores different patches for the same program, those patches are applied or uninstalled before they are executable), whereas Kraml maintains "a *new version* of the desired software application *and* at least one *older version* of the software application" (p. 5, ln. 5-8; emphasis added), either of which may be "copied into [a] second memory for execution" (col. 5, ln. 32-35). Aghera would thus render Kraml inoperative because

Aghera would dictate that one of Kraml's software versions would be deleted, or that only one executable version of the software could be maintained in storage at a time. Applicant therefore respectfully requests withdrawal of all rejections based on the combination of Aghera and Kraml.

Furthermore, Kraml discloses that the decision as to which software version to execute passes from the control center to the remote computer after the control center is notified an integrity error (see col. 6, ln. 47 – col. 7, ln. 10). Kraml further discloses that a “control center advantageously receives the message indicating that version n+1 has crashed and decides if version n+1 should be re-booted, or, alternatively, if a roll-back version n should be initiated” (col. 7, ln. 26-29). As discussed above, Aghera discloses multiple patches but only a single executable software program. Therefore, Applicant asserts that neither Kraml nor Aghera, alone or in combination, disclose or contemplate, “transferring, via radio frequency (RF) communication, software directly to a software-defined radio device from a software server to create transferred software, said software server remotely located with respect to said software-defined radio device, wherein said *transferred software is another version of software currently running in said software-defined radio device*, and wherein said transferred software is stored in at least a portion of a data store associated with said software-defined radio device; sending an instruction via RF communication directly to said software-defined radio device identifying said *transferred software or said software currently running in said software-defined radio device as a selected software application* to be loaded by said software-defined radio device in response to a restart of said software-defined radio device; and *automatically switching from said selected software application to a different version of said selected software application, wherein the different version is selected based on a detected error*” as recited in amended claim 1 (emphasis added). For at least these reasons, Applicant respectfully requests allowance of claim 1.

Furthermore, dependent claims 3 and 6-15 variously depend from independent claim 1, so Applicant asserts that dependent claims 3 and 6-15 are allowable over the cited references for the reasons set forth above for differentiating independent claim 1, in addition to their own unique features.

Similarly, and with respect to claim 27, Applicant asserts that neither Kraml nor Aghera, alone or in combination, disclose or contemplate, “a processor configured to . . . automatically revert, without an instruction from said software server, from said selected software to a different

software version responsive to at least one of said selected software encountering an error which causes said software-defined radio device to stop functioning properly, or exceeding a predetermined number of attempts to successfully load said selected software”, as recited in claim 27.

Furthermore, dependent claims 28 and 30-36 variously depend from independent claim 27, so Applicant asserts that dependent claims 28 and 30-36 are allowable over the cited references for the reasons set forth above for differentiating independent claim 27, in addition to their own unique features.

In addition, and with respect to claim 16, Aghera teaches against the present claims because Aghera teaches that only one executable version of the software is maintained in storage (see, e.g., para. 55). Applicant's claims, on the other hand, recite, for example “wherein said *transferred software* is *another version of software currently running in said software-defined radio device* . . . receiving, via radio frequency (RF) communication directly from said software server, an instruction at said software-defined radio device identifying said transferred software or said software currently running in said software-defined radio device as *a selected software application to be loaded* by said software-defined radio device in response to a restart of said software-defined radio device” (claim 16; emphasis added).

Additionally, Kraml discloses that “*first memory* is used to store *several versions* of the software application before they are copied *into second memory for execution*” (col. 5, ln. 32-35; emphasis added). “Second memory is advantageously a volatile memory (e.g., high-speed SRAM, etc.) that is of sufficient capacity to store any *one version* of the software application.” (col. 5, ln. 27-30; emphasis added). Therefore, Applicant asserts that neither Kraml nor Aghera, alone or in combination, disclose or contemplate, “wherein said *transferred software* is an *another version* of *software currently running in said software-defined radio device*, and wherein said software currently running in said software-defined radio device is stored in a *first non-volatile data store area*; storing said transferred software in a *second non-volatile data store area distinct from said first non-volatile data store area*; receiving, via radio frequency (RF) communication directly from said software server, an instruction at said software-defined radio device identifying said transferred software or said software currently running in said software-defined radio device as a *selected software application to be loaded* by said software-defined radio device in response to a restart of said software-defined radio device...” as recited

in claim 16 (emphasis added), and as similarly recited in claim 39. For at least these reasons, Applicant respectfully requests allowance of claims 16 and 39.

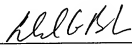
Furthermore, dependent claims 17-22 and 25-26 variously depend from independent claims 16, so Applicant asserts that dependent claims 17-22 and 25-26 are allowable over the cited references for the reasons set forth above for differentiating independent claims 16 and 39, in addition to their own unique features.

For example, with respect to claim 17, Kraml discloses that the decision as to which software version to execute, passes from the control center to the remote computer after the control center is notified an integrity error (see col. 6, ln. 47 – col. 7, ln. 10). Therefore, Kraml does not disclose or contemplate, alone or in combination with the cited references, “*automatically* reverting from said selected software application to a different software version *without an instruction from said software server*” as recited in claim 17 (emphasis added). For at least that additional reason, Applicant respectfully requests allowance of claim 17.

In view of the above remarks, Applicant respectfully submits that all pending claims properly set forth that which Applicant regards as their invention and are allowable over the cited references. Accordingly, Applicant respectfully requests allowance of the pending claims. The Examiner is invited to telephone the undersigned at the Examiner’s convenience, if that would help further prosecution of the subject application. The Commissioner is authorized to charge any fees due to Deposit Account No. 19-2814.

Respectfully submitted,

Dated: 7/13/09

  
\_\_\_\_\_  
David G. Barker  
Reg. No. 58,581

**SNELL & WILMER L.L.P.**  
400 E. Van Buren  
One Arizona Center  
Phoenix, Arizona 85004  
Phone: 602-382-6376  
Fax: 602-382-6070  
Email: dbarker@swlaw.com